

AMENDMENTS TO THE CLAIMS

1. (Canceled)

2. (Currently amended) The method according to ~~claim 1~~ claim 27, further including receiving as input ~~said~~ event types, business component types, and dependency types associated with a business domain.

3. (Currently amended) The method according to ~~claim 1~~ claim 27, further including receiving as input rules that describe how ~~an~~ a given event affects a specified business component.

4. (Currently amended) The method according to ~~claim 1~~ claim 27, further including receiving as input rules that describe when a change in a business component triggers an event.

5-7. (Canceled)

8. (Currently amended) The method according to ~~claim 1~~ claim 27, wherein the ~~business dependency model~~ definition includes predefined dependency type semantics.

9. (Currently amended) The method according to claim 8, wherein said dependency type semantics include a mandatory logical operator that logically couples one or more source components of the dependency to one or more targets of the dependency and sets the targets to a worst state of the source components.

10. (Currently amended) The method according to claim 8, wherein said dependency type semantics include an "N out

of M" logical operator, wherein N is less than M, that logically couples M source components of the dependency to one or more targets of the dependency and sets the targets to ok if at least N of the source components are ok and otherwise sets the targets to "fail".

11-26. (Canceled)

27. (New) A method for processing information, comprising:

in a system comprising one or more processors, providing an active dependency integration unit, comprising a first program module that receives as input first events for processing together with a definition of dependencies between business components in a business model in order to monitor a propagated impact between the business components;

providing in the system a situation management unit, comprising a second program module that detects situations comprising specified combinations of second events and conditions;

receiving in the active dependency integration unit a first event relating to at least a first business component;

responsively to the first event and to the dependencies, propagating a change to at least a second business component;

passing a second event indicative of the change to the situation awareness unit;

responsively to the second event, detecting a situation in the situation awareness unit;

responsively to the situation, conveying a third event from the situation awareness unit to the active dependency integration unit; and

IL920030025US1

outputting a functional state of the business model
responsively to at least the third event.